

# Is the Complicated ECC Array Necessary to Data Caches?

**Eui-Young Chung**

School of Electrical and Electronic Engineering,  
Yonsei University, Seoul 120-749, Korea

**Cheol Hong Kim**

School of Electronic and Computer Engineering,  
Chonnam National University,  
Kwangju 550-757, Korea

**Hyo-Joong Suh**

School of Computer Science and Information Engineering,  
The Catholic University of Korea, Gyeonggi-do 420-743,  
Korea

**Sung Woo Chung\***

Division of Computer and Communication Engineering,  
Korea University, Seoul 136-713, Korea  
Corresponding Author\*

**Abstract** - We propose a power-aware ECC (Error Correction Code) register without hurting the reliability of data caches. The ECC register replaces the complicated ECC array used in traditional data caches. While the traditional ECC in the ECC array is dedicated to one cache line, the proposed ECC register is shared by all the cache lines, which significantly reduces leakage power consumed in the ECC array. The simulation results show that the proposed ECC register consistently saves 8.5% of total power consumption in the 32KB data cache. On occurrence of a soft error, all the cache lines and the ECC register should be read to correct the soft error. However, the performance overhead from reading all the cache lines is negligible, since the Soft Error Rate (SER) is  $1.00E-10$  in the 70nm technology.

**Keywords:** Error Correction Code, Error Detection Code, Data Cache, Soft Error

## 1 Introduction

Soft error is a temporal malfunction caused by alpha particles, not leaving any permanent damages in the circuit. The SER (Soft Error Rate) is getting higher, as process technology is scaled down. The protection in the memory component is more crucial compared to the logic [4], since the soft error in the memory component is more complicated to be recovered. To protect the system from the soft errors, error recovery techniques have been proposed [1][2][5]. A representative example is a parity bit that is widely used for error detection in caches, memories, buses, and so on. Parity bits detect a single error but can not correct the error. When re-transmission or re-fetching is possible, a set of parity bits, which is an error detection technique, is a cost-effective solution. However, when data can not be recovered by re-transmission or re-fetching (eg. Data in data caches or in memories), error detection is not enough. In this case, it should be restored by an error correction technique. Hence, most data caches and memories adopt Error Correction Code (ECC) to be robust to soft errors.

Parity bits can not detect double bit error and ECC can detect double bit error. However, we concentrate on single bit error, since double bit error can be transformed into multiple single bit error by bit-interleaved parity bits and SER of double bit error is too rare [2].

## 2 ECC Array vs. ECC Register

In this paper, we focus on the ECC of a data caches. Since the access latency to a data cache is critical to performance, the ECCs are checked only when parity bits detect an error. Thus, there is no need to look up ECC array unless error is detected by parity bits. However, the ECC array accounts for substantial area of data caches, as shown in Fig. 1. Accordingly, leakage power from the ECC array is also substantial. Our goal is to reduce the leakage power consumed in ECC. We propose a simple ECC register shown in Fig. 2, instead of the complicated ECC array. In the traditional ECC array, each cache is corrected by referencing to the dedicated ECC. In the proposed ECC register, however, all the cache lines of the data cache are read and a soft error is corrected when parity bits detect a soft error. To enable the proposed technique, all the cache lines should be accessed whenever there is a soft error. Please note that the SER is lower than  $10E-10$  in 70nm technology. Thus, the performance degradation is only 0.00002% in worst cases, which is never serious. Whenever a write operation is issued, the ECC register should be updated, which might increase dynamic power consumption. In this case, the ECC register is updated by Exclusive-Oring the previous cache line, the current cache line and the previous ECC register. In other words, whole cache lines do not have to be accessed in case of a write operation. Please note that the ECC array should be updated every time there is a write even in the traditional caches.

There are several alternatives to Fig. 2. There could be more than one ECC register:  $n$  ECC registers accounts for  $1/n$  of total cache lines. Determining the number of ECC registers, there is a tradeoff between power consumption and performance degradation. Since the performance degradation is negligible in the cache with one ECC register, we only focus on one ECC register.

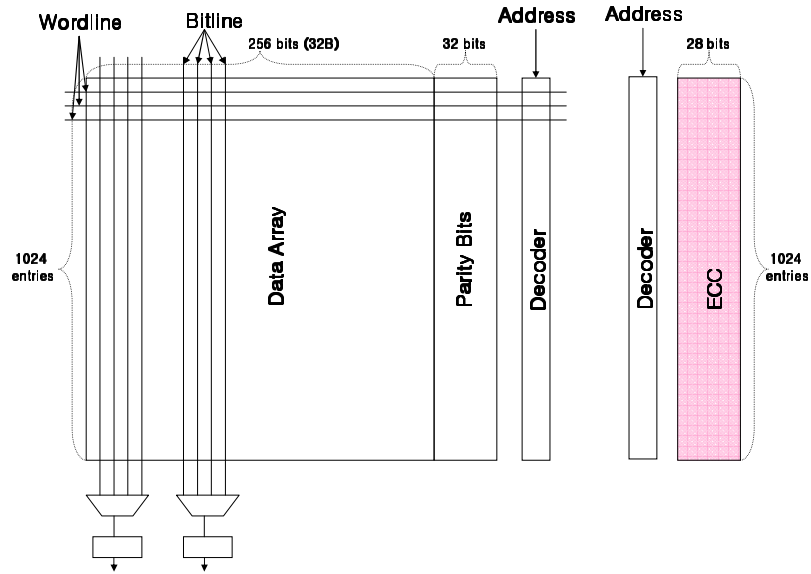


Fig. 1. Data Array and ECC Array in Data Cache

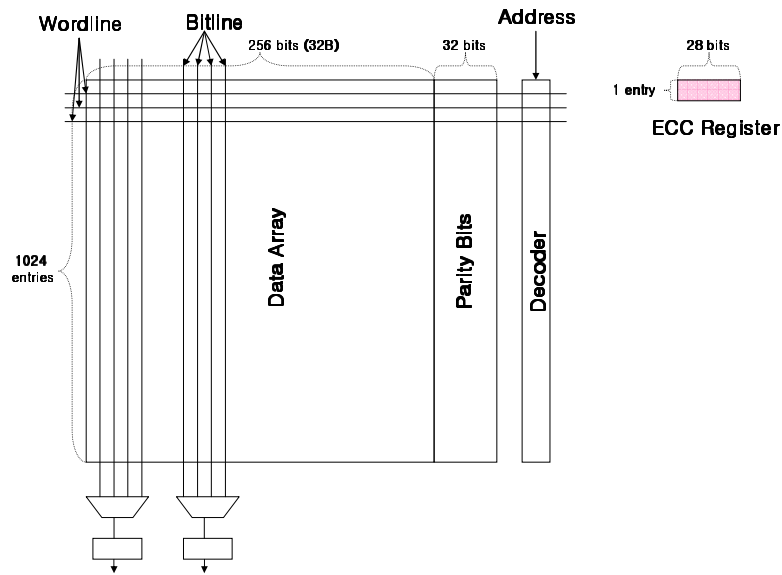


Fig. 2. Data Array and ECC Register in Data Cache  
(One ECC register accounts for all cache lines)

### 3 Simulation Results

This Power consumption is evaluated based on Cacti 4.0 [6] released by HP Labs. We assume the process technology is 70 nm and the target data cache is 32KB with 32B cache line. According to the parameters such as supply voltage, temperature, and, process technology, the SER

varies. Thus, Fig. 3 shows average power consumption excluding dynamic power consumption of the data array with difference SERs. Consistently, leakage power is reduced by 8.7%. Additional dynamic power consumption incurred by reading all the cache lines is negligible, since the SER is lower than  $1.00E-10$ .

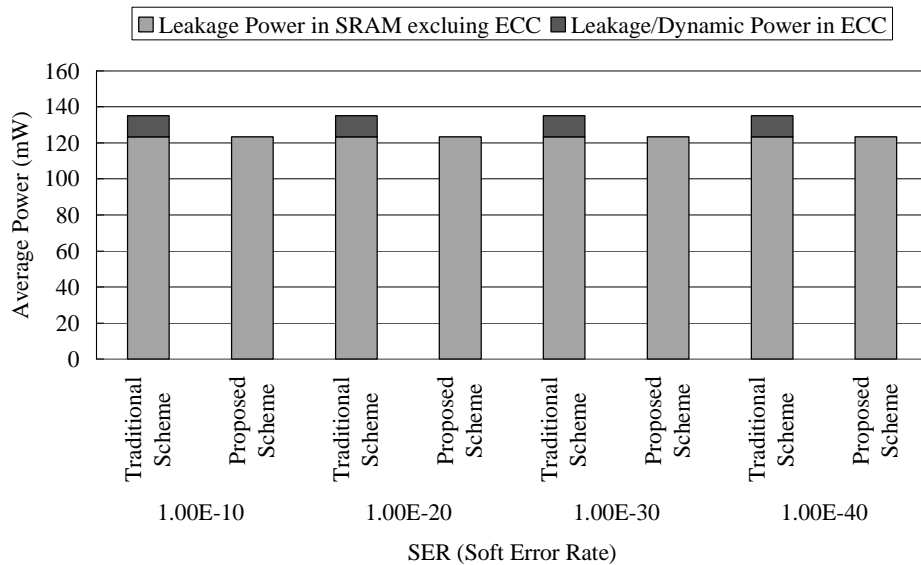


Fig. 3. Average Power Consumption with Different SER

## 4 Conclusions

One soft error in a data cache breaks down the whole system. To maintain the reliability of the system, the ECC is necessary to be robust to soft errors. In this paper, we proposed a simple ECC register that replaces the complicated ECC array in traditional data caches. Regardless of scaling down of process technology, the proposed technique consistently reduces the leakage energy 8.7% with negligible (lower than 0.00002%) performance overhead. Especially, the proposed technique is beneficial to embedded processors [3], where power consumption is crucial.

## Acknowledgments

This research is supported by a Korea University Grant.

## References

- [1] L. Li, V. Degalahal, N. Vijaykrishnan, M. Kandemir, and M. J. Irwin, "Soft Error and Energy Consumption Interactions : A Data Cache Perspective", In Proceedings of International Symposium on Low Power Electronics and Design, August, 2004.
- [2] S. S. Mukherjee, J. Emer, and S. K. Reinhardt, "The Soft Error Problem: An Architectural Perspective", In Proceedings of International Symposium on High-Performance Computer Architecture, Feb. 2005
- [3] V. Narayanan and Y. Xie, "Reliability Concerns in Embedded System Designs", IEEE Computer, vol.39 no.1, pp.118-120, Jan. 2006
- [4] N. Oh, P. Shirvani, and E. J. McCluskey, "Error Detection by Duplicated Instructions in Super-Scalar Processors", IEEE Trans. Reliability, 51(1):63-75, 2002.
- [5] F. Qin, S. Lu, and Y. Zhou, "SafeMem: Exploiting ECC-memory for Detecting Memory Leaks and Memory Corruption During Production Runs", In Proceedings of International Symposium on High-Performance Computer Architecture, Feb. 2005
- [6] D. Tarjan, S. Thoziyoor, and N. Jouppi, "Cacti 4.0", Tech Report:HPL-2006-86, available in <http://quid.hpl.hp.com:9081/cacti/>, HP Labs, 2006